

# Work Files

This section describes work files and valid formats that apply to the unload, load and scan functions of the Object Handler.

Object Handler functions invoked in a local environment will only process objects from this local environment, with a work file located in the current local file system. Object Handler functions invoked in a remote environment will only process objects from the same remote environment. The work file used for the load or unload function is located in the same remote environment or in your local file system.

See also *General in Settings - Options*, the section *Work Files* in the *Operations* documentation, the statement `DEFINE WORK FILE` in the *Statements* documentation and the profile parameter `WORK` in the *Parameter Reference* documentation.

This section covers the following topics:

- Work File Assignment
  - Work File Format
- 

## Work File Assignment

This section covers the following topics:

- Local Environments
- Remote Environments

### Local Environments

The following table lists the work files used by the Object Handler in local environments.

<b>File</b>	<b>Explanation</b>
Work File 1	Used for the unload, load and scan functions.  Contains the data unloaded.  See also <i>Transfer Work File</i> in <i>Tools</i> .
Work File 3	Used for internal reports.  Contains scan and find results.
Work File 4	Used if the option <b>Write report</b> (see <i>Settings - Options</i> ) is set. <b>Write report</b> is the default setting for object processing.  Contains report data.
Work File 5	The target file for the Adabas FDTs (Field Definition Tables) loaded.
Work File 6	Used for the load function if the option <b>Write restart information</b> (see <i>Restart Load</i> in <i>Functions</i> ) is set.  Contains restart information data.
Work File 7	An internal work file.
Work File 9	An internal work file.
Work File 10	Used if the trace mode is set.  See also <i>Traces</i> in <i>Tools</i> .
Work Files 11 to 15	Internal work files.

## Remote Environments

The following table lists the work files used in remote environments.

File	Location	Explanation
Work File 1	local system	<p>If the option <b>Local work file</b> is set (see <i>Settings - Options</i>), this work file is used for the unload, load and scan functions. It contains the data processed.</p> <p>Additionally, in mainframe environments, this work file is used to transfer work files from the local environment to the server and vice versa as described in <i>Transfer Work File in Tools</i>.</p>
Work File 1	server system	<p>If the option <b>Local work file</b> is <i>not</i> set (see <i>Settings - Options</i>), this work file is used for the unload, load and scan functions. It contains the data processed.</p> <p>Additionally, in mainframe environments, this work file is used to transfer work files from the local environment to the server and vice versa as described in <i>Transfer Work File in Tools</i>.</p> <p>Note that in a mainframe environment work files must be specified properly. See also <i>Natural User Access Method for Print and Work Files in Configuring Natural</i> in the documentation <i>Natural Operations for Mainframes</i>.</p>
Work File 3	local system	An internal work file.
Work File 9	local system	An internal work file.
Work Files 11 to 15	local system	Internal work files.

## Work File Format

There are two file formats for unloading objects in the source environment into work files and for loading them from work files into the target environment: an internal format and the Transfer format. Work files must be of internal format to transfer binary data. Work files must be of Transfer format to transfer text data.

This section covers the following topics:

- Internal Format
- Transfer Format

### Internal Format

The internal format is an internal record layout for work files that are used to transfer Natural sources and cataloged objects, error messages, Natural command processor sources, Adabas FDTs and non-Natural objects from one environment to another.

With the internal format activated, Natural objects are read from the source environment and written to a Natural work file by using the unload function of the Object Handler. This work file can be transported to another environment with standard file transfer services. In the target environment, the objects can then be read from the work file and loaded into the local file or database system with the load function of the Object Handler.

To transfer objects between identical platforms, use work files of internal format. Use portable work files of internal format if you want to transport objects between different UNIX, OpenVMS or Windows platforms, for example, from a little-endian machine to a big-endian machine. See also **Portable work file** in *Settings - Options, Portable Natural Generated Programs (Programming Guide)* and `DEFINE WORK FILE` (*Statements* documentation).

The Object Handler uses internal format by default. When using the internal format (**Transfer format** check box *not* selected), Work File 1 must be of binary format. To achieve this, omit the file extension or use the file extension `.sag`.

**Notes:**

1. Work files created by the utility SYSPAUL must be processed in internal format.
2. For remote environments: Work files created by the utility NATUNLD on the server, must be processed in internal format. The work files must be created on a server of the same platform where NATUNLD was applied.

## Transfer Format

See also **Transfer format** in the section *Settings - Options*.

The Transfer format is a general record layout for work files that contain load or unload data. This format is platform-independent and can be used to transfer the sources of Natural objects, Natural command processor sources, error messages and Adabas FDTs (Field Definition Tables) from one hardware platform to another and between Windows and mainframe, UNIX or OpenVMS platforms.

With the check box **Transfer format** selected, the unload function of the Object Handler reads Natural objects from a hardware platform and then restructures them.

Formatted records are written to a Natural work file that can be transported to another platform with standard file transfer services. On the target platform, the load function of the Object Handler then reads the objects from the work file and loads them into the local file or database system. The objects read from the work file are restructured according to the structure of the new hardware platform.

## Specifying Work Files in Local Environments

If Transfer format is specified (check box **Transfer format** selected), Work File 1 must be of text (ASCII) format. To achieve this, a file extension must be used, but not the file extension `.sag`. If Transfer format is *not* specified, Work File 1 must be of binary format. To achieve this, omit the file extension or use the file extension `.sag`.

## Specifying Work Files in Remote Environments

Work File 1 must be defined in the server environment. If Transfer format is specified (check box **Transfer format** selected), Work File 1 contains data of text (ASCII) format. If Transfer format is *not* specified, Work File 1 contains data of internal format.

## Handling Sources in Unicode/UTF-8

Transfer format is also used to unload or load sources of Natural objects in Unicode/UTF-8 (Universal Transformation Format, 8-bit form). If you specify the corresponding unload option (`WORKFILETYPE` set to `UTF-8` in command mode or **Unicode work file** in menu mode), all object sources will be unloaded into a work file in UTF-8. If you specify the corresponding load option (`LOAD-CODE-PAGE` in command

mode or **Load code page** in menu mode), all object sources in UTF-8 will be converted with the specified code page when they are loaded into a Natural system file.

### **Work Files from SYSTRANS**

Work files created by the utility SYSTRANS must be processed in Transfer format. Work files that contain object sources encoded in UTF-8 cannot be processed with SYSTRANS.